

Controlling Contrastive Self-Supervised Learning with Knowledge-Driven Multiple Hypothesis: Application to Beat Tracking

Antonin Gagnéré¹ Slim Essid¹ Geoffroy Peeters¹

Abstract

Ambiguities in data and problem constraints can lead to diverse, equally plausible outcomes for a machine learning task. In beat and downbeat tracking, for instance, different listeners may adopt various rhythmic interpretations, none of which would necessarily be incorrect. To address this, we propose a contrastive self-supervised pre-training approach that leverages multiple hypotheses about possible positive samples in the data. Our model is trained to learn representations compatible with different such hypotheses, which are selected with a knowledge-based scoring function to retain the most plausible ones. When fine-tuned on labeled data, our model outperforms existing methods on standard benchmarks, showcasing the advantages of integrating domain knowledge with multi-hypothesis selection in music representation learning in particular.

1. Introduction

Representation learning has proven effective for extracting meaningful features from complex data, with a recent focus on Self-Supervised Learning (SSL). SSL methods rely on pretext tasks—such as predicting masked inputs (Devlin et al., 2019; He et al., 2022) or contrasting augmented views of the same sample (van den Oord et al., 2019; Chen et al., 2020)—to learn general-purpose representations without labeled data. This flexibility reduces the need for costly annotations and expands the applicability of learning-based approaches to domains with limited labeling resources.

This work proposes to control SSL contrastive pre-training using multiple knowledge-driven hypotheses in the process of mining the positive and negative samples that will be considered in the contrastive scheme. We name this framework Knowledge-Driven Multiple Hypothesis Learning (KD-MHL).

¹LTCI Télécom paris. Correspondence to: Antonin Gagnéré <antonin.gagnere@telecom-paris.fr>.

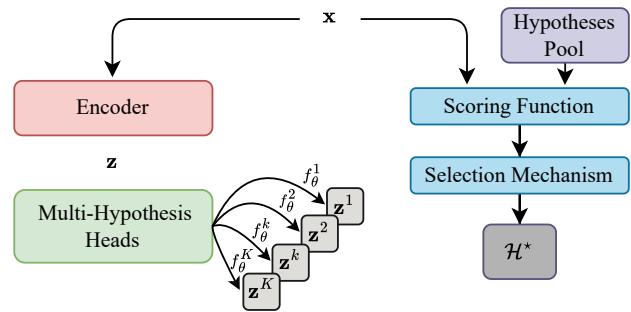


Figure 1. Overview of our SSL framework based on KD-MHL. The input sample x is encoded by g_θ into a representation z which is further projected by K heads f_θ^k corresponding to multiple hypotheses \mathcal{H}_k from a pool \mathcal{H} . Hypotheses are driven by the knowledge of the domain and lead to specific strategies for sampling anchors, positives, and negative samples within a contrastive framework. A function h_k scores each hypothesis. These scores are used by a mechanism s which selects the n winning hypotheses. At each step, the encoder is trained considering only the winning hypotheses (i.e. considering only the loss contributions from the winning heads).

KD-MHL incorporates ideas from Multiple Choice Learning (MCL), a learning paradigm tailored for ambiguous tasks, where multiple outputs are estimated by an ensemble of prediction heads plugged at the output of a deep network. Building on available annotations, various update mechanisms have been proposed to train the set of winning predictors (Guzmán-rivera et al., 2012; Rupprecht et al., 2016; Lee et al., 2016; Makansi et al., 2019; Letzelter et al., 2023). Resilient MCL (Letzelter et al., 2023), in particular, jointly trains both the winning hypothesis and the scoring functions associated with each predictor. Without having access to annotations one cannot rely on trainable scoring heads. Therefore KD-MHL uses a scoring function, based on prior-knowledge, which estimates how likely it is for a training sample to correspond to a given hypothesis. We instantiate our general framework to learn representations for musical rhythm analysis tasks, and show that they are highly effective for the problem of automatic beat and downbeat tracking, a key task in Music Information Retrieval (MIR). It aims to identify the temporal locations

of beats and downbeats in musical excerpts. The task is particularly challenging, both owing to the immense diversity of musical genres and the ambiguity inherent in the definition of ground-truth beat/downbeat labels, where different hypotheses may sometimes be made about correct beat positions by different human annotators.

Additionally, we explore self-training as a limit case of our framework, where the selection mechanism is nearly always correct. Models pre-trained in this way achieve state-of-the-art performance on most benchmark datasets, often surpassing previous systems by 2% in reference evaluation metrics.

Our key contributions are the following:

1. We introduce a new SSL framework that leverages multiple hypothesis learning to define its contrastive scheme, accounting for task ambiguity and leading to more powerful representations which readily accommodate multiple possible outcomes during downstream task adaptation.
2. We instantiate this framework to learn representations for musical rhythm analysis tasks and successfully apply them to beat and downbeat tracking.
3. We introduce a self-training variant of our approach, which achieves state-of-the-art beat and downbeat tracking performance on almost all reference benchmark datasets.

2. Related Work

Multiple Choice Learning Introduced in (Guzmán-rivera et al., 2012), MCL is a framework that employs multiple prediction heads to address tasks with inherent ambiguity. (Lee et al., 2016) extended this approach to deep learning, where heads are typically trained a Winner-Takes-All (WTA) strategy, updating only the head that produces the best prediction. To mitigate overconfidence, (Rupprecht et al., 2016) proposed a variant that also updates non-winning heads with scaled gradients. (Makansi et al., 2019) propose an evolving top- n WTA strategy, updating the top n predictions instead of just one. To improve uncertainty modeling, (Letzelter et al., 2023) incorporated a scoring function trained alongside the heads to estimate the probability of a hypothesis being among the winners.

Self-Supervised Learning Most common techniques for self-supervised learning leverage Siamese networks and rely on pairs of positive samples that share semantic information (Hadsell et al., 2006). These pairs are artificially created from unlabeled inputs by masking them (Assran et al., 2022; Bao et al., 2022; Assran et al., 2023) or by applying semantic-preserving data augmentations to them (Chen et al., 2020; Grill et al., 2020; Bardes et al., 2022; Zbontar et al., 2021). The Siamese architecture is then trained to

map those pairs to close locations in a latent space.

However, without additional constraints, the network would discard all information from the input and return always the same output. This phenomenon is called *representation collapse*. To prevent this, the most common strategy is to sample negative samples and repel them via a contrastive loss (Chen et al., 2020). Another approach is to explicitly penalize collapsed solutions by incorporating additional loss terms, as proposed in (Wang & Isola, 2020; Zbontar et al., 2021; Bardes et al., 2022).

Finally, another strategy consists of breaking the asymmetry between the two branches of the Siamese network, usually leveraging a momentum encoder (Grill et al., 2020; Chen & He, 2021; Caron et al., 2021). These techniques, while not explicitly avoiding representation collapse, have been shown to prevent this phenomenon in practice.

Self-supervised learning and in particular contrastive learning have been widely used in the audio domain. (van den Oord et al., 2019) applies it to raw waveforms for autoregressive prediction in the latent space, and (Baevski et al., 2020) extends it to audio masked modeling, taking inspiration from (Devlin et al., 2019). Later, inspired by the success of SSL with Siamese networks in the computer vision domain, (Saeed et al., 2020) adapts SimCLR (Chen et al., 2020) to the audio domain by using chunks from the same audio clip as positive pairs. Similarly, most SSL approaches originally introduced for computer vision have been then successfully applied to the audio domain, notable examples being (Anton et al., 2022; Niizumi et al., 2022).

Self-Supervised Learning for musical applications In the music domain, contrastive learning has been successfully applied to music representation learning. (Spijkervet & Burgoyne, 2021) leverage musically relevant data augmentations to create positive pairs, and (McCallum et al., 2022) scale up this approach to large music databases, achieving state-of-the-art performances in various downstream tasks. Many application of SSL to MIR relies on signal processing augmentations, that alter data with a semantic meaning e.g pitch shifting, time-stretching. (Gfeller et al., 2020) trained a network to predict the known difference between two pitch-shifted version of the same audio. This paved the way to other equivariant SSL (Dangovski et al., 2022) applications to various task such as pitch estimation (Riou et al., 2023), tempo detection (Quinton, 2022; Henkel et al., 2024; Gagné et al., 2024) or tonality estimation(Kong et al., 2024; 2025). Some works have explored self-supervised learning (SSL) for beat tracking. Zero-Note Samba (Desblancs et al., 2023) pre-trains a model by training an encoder to synchronize the latent representations of percussive and non-percussive stems. (Gagnere et al., 2024) is the closest to

our work. Hypothesizing a binary meter, they contrast latent representations separated by a power of two of Predominant Local Pulse (PLP) peaks.

3. Method

Beat-Tracking Data-driven approaches to beat tracking began with (Böck & Schedl, 2011), which employed bi-directional Long Short-Term Memory (LSTM) networks to process spectral features. This was subsequently improved by replacing LSTM with Temporal Convolutional Networks (MatthewDavies & Böck, 2019) and solving jointly beat and downbeat, as well as tempo detection (Böck et al., 2019; Böck & Davies, 2020). (Hung et al., 2022) applied a Spectral-Temporal Transformer (Lu et al., 2021) processing harmonic features and a temporal aggregation separately. Beat Transformer (Zhao et al., 2022) incorporates dilated self-attention to capture long-range dependencies alongside instrument-wise self-attention conducted along the stems of demixed audio. All these methods rely on Dynamic Bayesian Network (DBN) (Krebs et al., 2013a) to post-process activations. (Foscarin et al., 2024) achieved state-of-the-art performance by introducing shift-tolerant Binary Cross Entropy (BCE) to remove the DBN dependency.

An overview of our framework is presented in Figure 1. In the following, we detail its general formulation and key elements, especially how it models in a self-supervised manner different sampling strategies using an ensemble of output heads.

3.1. General Framework

Given an input sample \mathbf{x} —which may be a sequence or a bag of samples from the training dataset \mathcal{X} —, we extract an intermediate representation \mathbf{z} with an encoder g_θ . Whereas standard contrastive learning approaches typically rely on data augmentation or sample mining to define positive and negative pairs, our method specializes \mathbf{z} into K different representations $\{\mathbf{z}^k\}$, using an ensemble of heads $f_\theta \triangleq (f_\theta^1, \dots, f_\theta^K)$. Each head f_θ^k corresponds to a domain-motivated *hypothesis* (driven by knowledge), \mathcal{H}_k , which specifies a strategy for sampling positives and negatives within the sequence or bag of samples. We denote the set of all hypotheses by \mathcal{H} .

This formulation is motivated by the inherent ambiguity that may arise when predicting an output for a given task. Rather than forcing the model to commit to a single valid interpretation we exploit multiple heads that enable it to account for different possible outcomes. To exploit such a method, one needs to have prior domain knowledge to define a set of plausible hypotheses but also to design the associated mining strategies.

To guide which hypotheses should apply to \mathbf{x} , we introduce

a scoring function $\mathbf{h} \triangleq (h_1, \dots, h_K) : \mathcal{X} \rightarrow \mathbb{R}^K$ that measures the “compatibility” between the data sample and each hypothesis k . Let $\mathcal{T}_k = \{A_k, P_k, N_k\}$ denote, respectively, the sets of anchors, positives, and negatives under hypothesis k . A selection mechanism $\mathbf{s} : \mathbb{R}^K \rightarrow 2^{\{1, \dots, K\}}$ outputs the subset of winning hypotheses. The overall contrastive loss $\mathcal{L}(\mathbf{x})$ (used for SSL training) is then defined as the sum of the contributions under the selected hypotheses:

$$\mathcal{L}(\mathbf{x}) = \sum_{k \in \mathbf{s}(\mathbf{h}(\mathbf{x}))} \mathcal{L}(\mathbf{z}^k, \mathcal{T}_k), \quad (1)$$

By allowing multiple hypotheses to drive the choice of anchor, positive, and negative sets, we believe we help the model handle potentially ambiguous tasks.

3.2. Hypothesis selection

Given the score h_k computed for each hypothesis $\mathcal{H}_k \in \mathcal{H}$, the selection mechanism, \mathbf{s} , determines a suitable subset of hypotheses to retain: \mathcal{H}^* . We explored several selection strategies, including the WTA approach, where only the best-scoring hypothesis is selected and n -WTA where the n best hypotheses are retained:

$$\mathbf{s}_{n\text{-WTA}} = \arg \min_{\mathcal{S} \subset \{1, \dots, K\}, |\mathcal{S}|=n} \sum_{k \in \mathcal{S}} h_k \quad (2)$$

4. Instantiation on Musical Rhythm Analysis

In the following, we specify the set of hypotheses \mathcal{H} and their selection \mathbf{h}/\mathbf{s} for the case of rhythm analysis (beat and downbeat tracking). Figure 2 depicts the system adapted rhythm analysis.

4.1. Hypothesis definition

In the case of SSL, one does not have access to ground-truth output labels for training (here the beat/downbeat positions, for instance). Instead, one needs to solve a pretext-task, obtaining supervision from the input data itself. For rhythm analysis (especially beat and down-beat detection), we rely on audio signal processing techniques to get such supervision: for each audio sample \mathbf{x} we compute the so-called predominant local pulse (PLP) function to drive the selection of plausible output hypotheses. This scalar function captures the dominant rhythmic pulse at each time instant in an audio signal. It is computed by first deriving a sinusoidal kernel for each time position that best explains the local periodicity of an Onset Strength Function (OSF). The OSF measures the likelihood that a musically salient change (e.g., note onset) has occurred at each time point. All kernels are then accumulated over time using overlap-add synthesis.

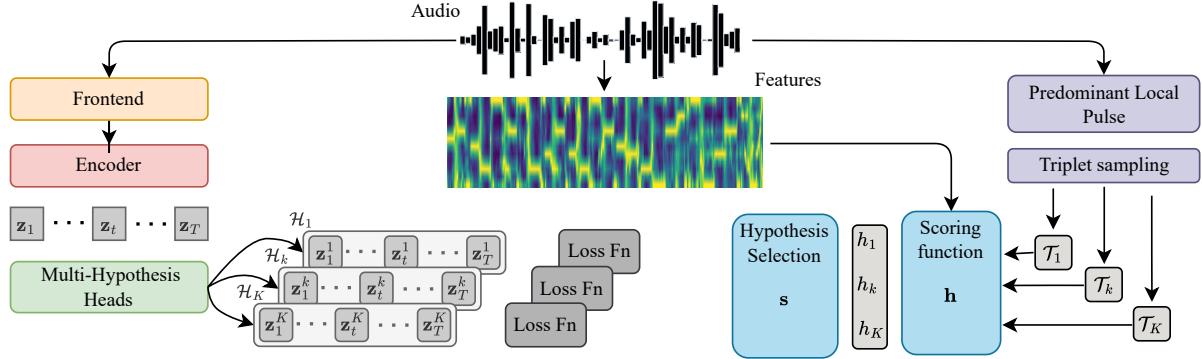


Figure 2. Instantiation of our SSL framework based on KD-MHL for musical rhythm analysis (beat and downbeat tracking). The input is a sequence \mathbf{x}_t that represents the audio signal over time t which is projected by g_θ into a sequence of \mathbf{z}_t . The objective is to train g_θ such that \mathbf{z} takes different values when t is a beat or not. This is achieved using contrast learning, sampling triplets (anchors, positive and negative times). Driven by knowledge, we create a pool of hypothesis $\mathcal{H}_k \in \mathcal{H}$ which correspond to possible metrical relationship between PLP peaks (which define the time units t) and beats; and therefore correspond to specific triplet samplings \mathcal{T}_k . Each \mathcal{H}_k is scored by h_k considering the audio features evolution under the given metrical relationship. These scores are used by a mechanism s which selects the n winning hypotheses. At each step, the encoder g_θ is trained considering only the n winning hypotheses (i.e. considering only the loss contributions from the n winning heads)

The result is a function like the one given in Figure 3, where each peak is a plausible candidate for a beat position.

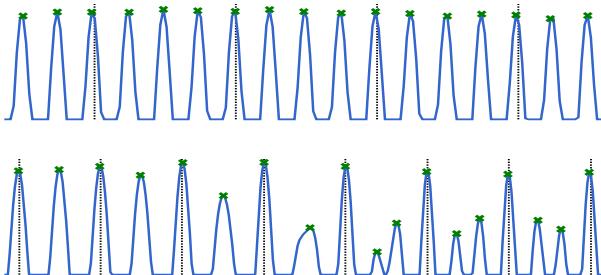


Figure 3. PLP function (blue) alongside beat annotations (dashed lines) and detected peaks (green crosses). The top plot display a (4,2) hypothesis, the bottom one displays phase shifting issue.

The sequence of peaks from the PLP function is represented as a set of time positions: $\mathcal{B} = \{b_i\}_{i=1}^B$, where B corresponds to the number of detected peaks. We make the assumption that these PLP peaks \mathcal{B} are aligned with the tatus.¹ Because of this, the actual beats form a subset of \mathcal{B} . However, the exact metrical relationship between these detected peaks and the true beats is unknown. In other words, one does not know if every peak corresponds directly to a true beat, or if the true beats only occur every ω peaks, $\omega \in \mathbb{N}$. Also one does not know where the first true beat

¹The tatum corresponds to the fastest regular rhythmic events that occur in a piece of music. A beat interval is an integer number of tatum intervals.

starts in the sequence of \mathcal{B} . To formalize this, we define a ratio $\omega \in \mathbb{N}$ and an initial phase $\phi \in \mathbb{N}$ such that the subset:

$$\mathcal{B}_{\omega,\phi} = \{b_{\omega k + \phi} \mid 1 \leq \omega k + \phi \leq B\} ; k \in \mathbb{N} \quad (3)$$

corresponds to the true underlying beat sequence. In the top part of Figure 3 the underlying ground-truth corresponds to the subset $\mathcal{B}_{4,2}$. We denote by Ω the set of ratios considered for ω . Given the problem definition, the set of possible phases for a given ω is $\Phi_\omega = \{\phi \mid 0 \leq \phi < \omega\}$. This yields a total of K hypotheses with $K = \sum_{\omega \in \Omega} \omega$:

$$\mathcal{H} = \{\mathcal{H}_{\omega,\phi} \mid \omega \in \Omega, \phi \in \Phi_\omega\}. \quad (4)$$

4.2. Scoring function

Given a set of hypotheses \mathcal{H} , our objective is to identify the underlying correct one/ones. To achieve this, we define for each k a contrastive scoring function $h_k(\mathbf{x})$ based on audio features extracted from \mathbf{x} . This function is designed to measure the mutual information between positive and negative elements chosen from \mathcal{B} with respect to hypothesis k . The intuition is that, in music, repetition often becomes more salient at moments corresponding to beats and/or downbeats. By leveraging this repetitive structure, the scoring function can effectively distinguish between hypotheses, prioritizing those that align with the rhythmic patterns inherent in the musical segment.

A hypothesis $\mathcal{H}_{\omega,\phi}$ defines a candidate subset $\mathcal{B}_{\omega,\phi}$. From this subset, we sample an anchor time step $A_{\omega,\phi}$, followed by n_p positive time steps $P_{\omega,\phi} = \{p_{\omega,\phi}^i\}_{i=1}^{n_p}$, selected from the remaining candidates in $\mathcal{B}_{\omega,\phi} \setminus A_{\omega,\phi}$. For the negative time steps $N_{\omega,\phi} = \{m_{\omega,\phi}^i\}_{i=1}^{n_n}$ we include i) easy

negatives, *i.e.*, time steps not aligned with PLP peaks; ii) hard negatives, *i.e.*, time steps that correspond to PLP peaks but do not belong to $\mathcal{B}_{\omega,\phi}$. We apply this procedure to each hypothesis $\mathcal{H}_{\omega,\phi}$, to obtain its associated set of triplets $\mathcal{T}_{\omega,\phi} = \{A_{\omega,\phi}, P_{\omega,\phi}, N_{\omega,\phi}\}$. For conciseness, we denote by $\mathcal{H} = \{\mathcal{H}_k\}_{k=1}^K$ the set of hypotheses and by $\mathcal{T}_k = \{A_k, P_k, N_k\}$ their associated triplets.

From the audio signal $\mathbf{x} = \{x_l\}_{l=1}^L$, we extract a sequence of feature vectors $\mathbf{V}^\mathbf{x} = [\mathbf{v}_1^\mathbf{x}, \mathbf{v}_2^\mathbf{x}, \dots, \mathbf{v}_T^\mathbf{x}] \in \mathbb{R}^{d_v \times T}$, where $\mathbf{v}_t^\mathbf{x}$ is the audio feature corresponding to time frame t . For each hypothesis \mathcal{H}_k , we define the contrastive scoring function

$$h_k(\mathbf{x}) = - \sum_{t_p \in P_k} \log \frac{\exp(\text{sim}(\mathbf{v}_{A_k}^\mathbf{x}, \mathbf{v}_{t_p}^\mathbf{x}) / \tau)}{\sum_{t_n \in N_k} \exp(\text{sim}(\mathbf{v}_{A_k}^\mathbf{x}, \mathbf{v}_{t_n}^\mathbf{x}) / \tau)}, \quad (5)$$

where τ is a temperature parameter and $\text{sim}(\mathbf{a}, \mathbf{b})$ denotes cosine similarity: $\text{sim}(\mathbf{a}, \mathbf{b}) = \frac{\mathbf{a}^\top \mathbf{b}}{\|\mathbf{a}\| \|\mathbf{b}\|}$.

4.3. Training strategy

Given the scores h_k we extract a subset of select hypotheses with s . For each selected hypothesis, $\mathcal{H}_k \in \mathcal{H}^*$ we compute the following loss:

$$\begin{aligned} \mathcal{L}_k(\mathbf{z}) &= \mathcal{L}_{\text{NT-Xent}}(\mathbf{z}, \mathcal{T}_k) \\ &= - \sum_{t_p \in P_k} \log \frac{\exp(\text{sim}(\mathbf{z}_{A_k}^k, \mathbf{z}_{t_p}^k) / \tau)}{\sum_{t_n \in N_k} \exp(\text{sim}(\mathbf{z}_{A_k}^k, \mathbf{z}_{t_n}^k) / \tau)}. \end{aligned} \quad (6)$$

5. Experiments

5.1. Datasets

For the self-supervised *pre-training* of our models, we used the Million Song Dataset (Bertin-Mahieux et al., 2011), a large collection of contemporary popular music tracks. We had access to audio excerpts of approximately 905,000 tracks. Following the explanations in Sections 5.3 and ??, we extracted a fixed set of 20-second audio chunks \mathbf{x} from the tracks, allowing up to five chunks per track. We only kept those with quasi-constant inter-PLP peak intervals. This ensures a controlled training set reducing PLP-extraction related errors. These chunks were selected from approximately 600,000 pieces, resulting in over 3 million unique chunks.

For *fine-tuning* our models, we used the Ballroom (Gouyon, 2006; Krebs et al., 2013b), Beatles (Davies et al., 2009), Hainsworth (Hainsworth & Macleod, 2004), Harmonix (Nieto et al., 2019), HJDB (Hockman et al., 2012), SMC (Holzapfel et al., 2012) and RWC-Popular (Goto et al., 2002; Goto, 2006) datasets. In addition, we kept the GTZAN dataset (Tzanetakis & Cook, 2002; Marchand & Peeters, 2015), as commonly done in previous works, to serve

as an unseen test set for evaluating generalization performance. While additional annotated datasets exist, they are not openly accessible.²

5.2. Network Architecture

The present work does not focus on designing a new deep network architecture. We therefore adopt the openly available model from (Foscarin et al., 2024) as our backbone. This model provides the current state-of-the-art results in beat and downbeat tracking.

Our pre-training model consists of *i*) a frontend, *ii*) an encoder network that outputs a sequence of representations, and *iii*) our proposed projection layers f_θ^k for each hypothesis k . The *input* is a 128-bin mel spectrogram where the magnitudes are scaled by $\ln(1 + 1000x)$. The *frontend*'s role is to integrate information across the 128 frequency bands into feature vectors. It comprises three blocks, each containing two Partial Transformers inspired by (Lu et al., 2023), which first process time, then frequency, followed by a 2D convolution, batch normalization, and GeLU activation. The first Partial Transformer treats each time step as a sequence across frequency bands, capturing harmonic structure, while the second operates across time steps for temporal modeling. Each block reduces the number of frequency bands while increasing channels, progressively refining the feature representation. A 2D convolution further processes the extracted features while preserving spatial structure. After three blocks, the output is reshaped and projected into a $T \times 512$ representation.

The *encoder* network consists of six transformer layers with 16 heads of dimension 32, using rotary positional embeddings (Su et al., 2024) and sigmoid gating (Bondarenko et al., 2023). The feed-forward dimension is set to 2048. The sequence is then processed by the hypothesis heads, which consist of K linear layers that project the 512-dimensional sequence into a 64-dimensional sequence, followed by RMS normalization. In total, the network contains 20.3 million parameters.

5.3. Training Setup

The input, \mathbf{x} , is a 20-s long chunk of audio sampled at 16 kHz. We turn it into a Mel-spectrogram, which is computed with a hop size of 20 ms (320 samples); this is therefore the time granularity of our representation. For PLP extraction, we first compute the OSF using the Superflux method (Böck & Widmer, 2013) and the same hop length (this is because

²While (Foscarin et al., 2024) provides pre-computed features as 22,050 Hz Log-Mel Spectrograms (LMS) for other datasets, their audio is not accessible. We attempted to get the audio back from the LMS using the Griffin-Lim algorithm but the resulting quality was insufficient. Therefore we chose not to include these datasets in our experiments, to avoid unfair comparisons.

Table 1. Beat and Downbeat detection results with single split fine-tuning. Metrics are computed on the GTZAN dataset. \star stand for the backbone from (Foscarin et al., 2024) and \dagger the one used in (Gagnere et al., 2024)

PRE-TRAINING METHOD	FINE-TUNING METHOD	BACK-BONE BONE	BEAT			DOWNBEAT			AVG
			F1	CMLT	AMLT	F1	CMLT	AMLT	
WTA	BCE-DBN	\star	88.7	81.0	92.4	75.8	71.7	87.5	82.9
WTA	ST-BCE	\star	89.2	79.8	90.4	76.9	63.5	78.6	79.7
2-WTA	BCE-DBN	\star	88.5	80.4	92.2	75.2	70.5	86.7	82.3
2-WTA	ST-BCE	\star	88.8	79.4	89.5	75.4	60.7	75.3	78.2
3-WTA	BCE-DBN	\star	89.2	82.1	92.8	75.3	71.7	87.5	83.1
3-WTA	ST-BCE	\star	88.9	79.9	89.7	75.9	61.9	77.6	79.0
3-WTA	BCE-DBN	\dagger	88.1	80.6	91.6	75.7	71.5	87.0	82.4
3-WTA	ST-BCE	\dagger	88.0	78.5	88.6	73.9	58.3	74.0	76.9
SELF-TRAINING	BCE-DBN	\star	89.6	82.6	92.5	78.3	74.7	88.2	84.3
SELF-TRAINING	ST-BCE	\star	89.6	81.8	91.8	77.5	67.0	80.8	81.4
(BÖCK & DAVIES, 2020)			88.5	81.3	93.1	67.2	64.0	83.2	79.6
(HUNG ET AL., 2022)			88.7	81.2	92.0	75.6	71.5	88.1	82.9
(ZHAO ET AL., 2022)			88.5	80.0	92.2	71.4	66.5	84.4	80.5
(FOSCARIN ET AL., 2024)			89.1 ± 0.3	79.8 ± 0.6	89.8 ± 0.4	78.3 ± 0.4	67.3 ± 0.8	79.1 ± 0.6	80.6
– LIMITED TO DATA OF (HUNG ET AL., 2022)			88.9 ± 0.1	79.9 ± 0.4	89.4 ± 0.2	75.5 ± 0.5	60.8 ± 1.2	75.5 ± 0.5	78.4

the PLP and input time frames should be aligned), followed by a high-pass filter.

We discard chunks where the distance between successive PLP peaks is not approximately constant. These variations may result from actual changes in the music (such as tempo shifts, the introduction of new elements, or silences), or from inconsistencies in the PLP. Such cases typically indicate a phase and/or ratio shift, as illustrated in Fig. 3, where the ratio shifts from 2 to 3. When this occurs, it becomes impossible to determine a fixed ratio and phase to explain the beat sequence from the PLP peaks, violating the assumption in equation 3. In practice, we discard chunks if the relative variation in inter-peak distances exceeds 20% between two successive peaks. This threshold is high enough to allow for slight tempo variations but prevents phase and/or ratio shift to occur.

Based on preliminary studies on annotated data, we restrict the set of considered ratios to $\Omega = \{1, 2, 3, 4\}$ resulting in a total of $K = \sum_{\omega \in \Omega} \omega = 10$ hypotheses. Indeed the PLP rarely detects rhythmic events going faster than four times the beat. Our scoring function h is computed for each hypothesis with Variable-Q chromagram features also computed with a 20ms hop size. We found that it was beneficial to stack input features with time-lagged copies of itself.

To improve scoring stability, we trained models on a fixed set of chunks $\mathbf{x} \in \mathcal{X}$. At each epoch e , we store the scores $h_k^e(\mathbf{x})$ for each chunk \mathbf{x} and each hypothesis $\mathcal{H}_k \in \mathcal{H}$. These scores depend on the random sampling of the anchor, positives and negatives of \mathcal{T}_k (4.2). We therefore use the running mean over all past epochs $h_k^{[0:e]}(\mathbf{x})$ as the final score for hypothesis selection. This aggregation reduces score fluctuations, ensuring a more stable selection function

and a smoother training process.

Our models were pre-trained for 200,000 steps in FP16 precision using 8 V100 GPUs with a global batch size of 128. Gradient clipping was applied to stabilize training. We used AdamW optimizer (Loshchilov & Hutter, 2019) alongside a cosine learning rate scheduler. The learning rate was linearly increased to 0.0005 over the first 32,000 steps and then gradually decreased for the remaining training steps.

5.4. Fine-tuning methods

After pre-training we discard our ensemble of heads f_θ and use directly the intermediate representations \mathbf{z}_t . These are projected, with a linear layer, into two scalar outputs that represent beat and downbeat activations respectively. We explored two different fine-tuning methods.

BCE-DBN: The first method trains these activations to minimize the BCE loss, with a target-widening strategy, the neighboring frames of a frame annotated as positive are also set as positive targets but with a lower weight of 0.5 (Böck & Davies, 2020). Once training is complete, the activations are post-processed using a DBN (Krebs et al., 2018) to produce the final beat and downbeat positions.

ST-BCE: The second approach (Foscarin et al., 2024) introduces a Shift-Tolerant Weighted BCE (ST-BCE) loss function to address i) class imbalance, ii) annotations’ imprecision and iii) the reliance on a DBN for post-processing. In the ST-BCE approach, positive examples are weighted by a factor α defined as the ratio of negative to positive frames, computed separately for beats and downbeats. Predictions are max-pooled over time before comparing them to ground-truth labels, ensuring that the strongest posi-

tive activation within a local window is considered, while negative labels near annotated beats are ignored. This reduces sensitivity to slight misalignment and prevents over-penalization of near-correct predictions. Formally the loss is defined as: $L_{st}(\mathbf{y}, \hat{\mathbf{y}}) = -\sum_t \alpha y_t \log(m_7(\hat{\mathbf{y}})_t) + (1 - m_{13}(\mathbf{y})_t) \log(1 - m_7(\hat{\mathbf{y}})_t)$ where m_l denotes max-pooling over l frames. Beat and downbeat detection are obtained by applying peak-picking on their respective activations functions.

In each setting, we trained both the *front-end* and *encoder* for 100 epochs using the AdamW optimizer (Loshchilov & Hutter, 2019) with a batch-size of 64. During the initial 3 epochs, the *encoder* weights were kept frozen. The learning rate was scheduled with a cosine annealing strategy, ramping up to 0.001 over the first 400 optimization steps and gradually decaying to 0 by the final epoch. We employ the same data augmentations as those of Beat This (Foscarin et al., 2024). Every track can be either pitch-shifted by a transposition factor sampled in [-5, +6] semitones or time-stretched with a factor in 20, 16, 12, 8, and 4% both faster and slower.

5.5. Evaluation metrics

We report the standard evaluation metrics commonly used in the literature (Davies et al., 2009; Davies & Böck, 2014). The F-measure considers a detected beat correct if it falls within a ± 70 -ms tolerance window around a ground-truth beat position. Additionally, we use continuity-based metrics, where a beat is valid only if the previous beat is also correct. The CMLt metric measures the proportion of beats correctly aligned with annotations at the expected metrical level under this constraint. The AMLt metric extends this by allowing metrical variations, such as double or half tempos or off-beat shifts. We computed these metrics using the `mir_eval` package (Raffel et al., 2014), following the convention of trimming beats occurring in the first 5 seconds and using default parameters.

5.6. Single-split fine-tuning

We split each dataset (except GTZAN that we keep for evaluation) into train and validation (7/8, 1/8) and use the union of the train splits to fine-tune our pre-trained models. We use the same splits as (Foscarin et al., 2024). This serves as an experiment to explore the behavior of the sampling mechanism s . We explored 3 different mechanisms: WTA, 2-WTA and 3-WTA. We also compare the backbone architecture of (Foscarin et al., 2024), to the one used in (Gagnere et al., 2024). The latter consists of a mel-spectrogram fed to a transformer encoder without rotary positional encoding.

We report the results in the upper part of Table 1. The lower part of the Table provides baselines from state-of-the-art

systems. It is important that, at the exception of (Foscarin et al., 2024) the performances are not obtained in the same conditions.

We see that the backbone of (Foscarin et al., 2024) consistently outperforms (Gagnere et al., 2024) under equivalent pre-training and fine-tuning configurations.

Using 3-WTA improves beat metrics compared to one- or two-WTA variants: 3-WTA + BCE-DBN achieves the highest Beat F1 (89.2), CMLt (82.1), and AMLt (92.8). Furthermore, a trade-off emerges, between using the BCE with DBN and ST-BCE with peak-picking echoing prior work (Foscarin et al., 2024). While BCE-DBN generally leads to superior continuity metrics (CMLt, AMLt), ST-BCE can lead to better F1. This trade-off is also evident in the downbeat results, where ST-BCE yields higher F1 (e.g., 76.9 for 1-WTA).

5.7. Cross-validation

In the second experiment, we split each dataset into train, validation and test (6/8, 1/8, 1/8), use the union of the trains to fine-tune our pre-trained models and the union of the tests for evaluation. The folds are changed following and 8-fold cross-validation. This is the methodology commonly used in the literature. Here also we used the same folds of (Foscarin et al., 2024).

The upper part of Table 2 presents the average beat and downbeat metrics for our pre-trained models, while the lower part shows the performance of existing systems. Additionally, Tables 4 and 5 in Appendix B provide detailed metrics for beat and downbeat, respectively. Cross-validation results are reported only for WTA and 3-WTA pre-training, as they demonstrated the best performance.

Using 3-WTA for pre-training leads to improved average performance across both fine-tuning strategies. Specifically, 3-WTA + BCE-DBN achieves an 87.4 average, slightly exceeding the 87.1 from WTA + BCE-DBN. We also observe particularly strong performance on Hainsworth (Beat: 89.0, Downbeat: 79.8). However, performance on RWC Popular (92.4 for both beat and downbeat) lags behind that of (Hung et al., 2022), which consequently lowers our overall average compared to their 88.0. Meanwhile, the 3-WTA + ST-BCE setup (85.4 average) is slightly below (Foscarin et al., 2024) (86.1); however, our fine-tuning is performed on a smaller training set (3144 vs. 4556 tracks), which likely accounts for the difference.

5.8. Self-Training

The previous results were obtained using the proposed multi-hypothesis contrastive pre-training approach. To establish an upper bound on performance, we investigate an advanced setting in which the correct hypothesis is consistently se-

Table 2. Average cross-validation scores for Beat (B) and Downbeat (DB) detection systems on reference datasets.

PRE-TRAINING	FINE-TUNING	BALLROOM		BEATLES		HARMONIX		HAINSWORTH		RWC POPULAR		SMC	AVG
		B	DB										
WTA	BCE-DBN	96.8	<u>95.2</u>	90.8	82.4	93.0	88.6	89.8	79.4	92.1	91.8	<u>58.3</u>	87.1
WTA	ST-BCE	<u>96.9</u>	93.6	90.9	77.7	93.2	85.0	89.0	72.2	91.3	88.8	<u>54.7</u>	84.8
3-WTA	BCE-DBN	96.8	95.0	<u>91.3</u>	<u>83.7</u>	<u>93.4</u>	<u>89.1</u>	89.0	<u>79.8</u>	92.4	<u>92.4</u>	58.2	<u>87.4</u>
3-WTA	ST-BCE	96.6	93.4	90.8	78.2	93.1	84.9	<u>90.3</u>	72.3	<u>92.6</u>	90.6	56.1	85.4
SELF-TRAINING	BCE-DBN	97.5	97.2	93.6	86.0	95.0	92.4	93.2	83.6	91.3	91.0	61.9	89.3
SELF-TRAINING	ST-BCE	97.8	95.7	94.2	82.0	95.3	88.4	93.2	77.3	91.1	89.3	60.3	87.7
(FOSCARIN ET AL., 2024)		97.0	94.1	91.6	81.5	93.2	85.9	88.9	72.9	93.3	90.0	58.4	86.1
(BÖCK & DAVIES, 2020)		95.7	93.0	81.4	-	90.4	80.8	89.7	76.3	-	-	55.3	60.2
(HUNG ET AL., 2022)		95.6	94.4	92.6	84.9	95.0	90.3	88.5	78.5	94.4	94.8	59.4	88.0
(ZHAO ET AL., 2022)		96.3	95.1	-	-	93.9	89.3	88.7	76.7	-	-	56.2	54.2

lected during training. However, in the context SSL, ground-truth annotations are not available, making it infeasible to directly determine the correct hypothesis.

To approximate this ideal condition, we employ a self-training strategy based on pseudo-labeling. Specifically, we generate pseudo-labels as surrogate beat annotations, which enables the selection of the most relevant hypothesis. This approach is equivalent to the WTA pre-training paradigm, where the scoring function s always identifies the correct underlying hypothesis. Given these pseudo beat positions, we apply the same sampling procedure described in Section 4.2, with $\omega = 1$. Unlike the training method described in 3.1, we do not use the ensemble of hypothesis-specific projection heads f_θ . Instead, the contrastive loss is computed directly on the general sequence of learned representations, z_t .

We report the system accuracy in single split (middle part of Table 1) and averaged metric for cross-validation fine-tuning (middle part of Table 2). Detailed metrics for 8-fold cross-validation are gathered in B.

In Table 1, self-training + BCE-DBN achieves an 84.3 average on GTZAN, outperforming all other listed methods. This gain comes from notable improvements in both beat detection (+0.5 F1 and +1.4 CMLt) and downbeat detection (+2.7 F1 and +3.2 CMLt).

In Table 2, the same approach reaches the highest average score 89.3 — which is +1.3 above (Hung et al., 2022) and +3.2 above (Foscarin et al., 2024). The largest gains appear on Hainsworth (+3.5 average beat metrics, +5.1 average downbeat metrics) and SMC (+2.5), a particularly challenging dataset. RWC Popular sees lower performance (-3.1 and -3.8), but on the remaining datasets both fine-tuning methods match or surpass prior state-of-the-art methods. Overall, these results confirm that self-training provides a robust performance boost for beat and downbeat detection.

6. Conclusion

We proposed Knowledge Driven Multi Hypothesis Learning to guide contrastive SSL. In this framework we rely on domain knowledge to score, at each step and for each sample, a set of hypotheses, select the n-winning ones, and use those for training. We instantiated this framework for the task of rhythm analysis and explored different hypothesis selection mechanisms for pre-training. We fine-tuned the model with annotated data and demonstrated state-of-the-art performance on several benchmarking datasets, confirming the effectiveness of our pre-training. Additionally, we explored self-training as an advanced setting of our framework. Pre-training a model this way yielded state-of-the-art performance on most benchmarks, often surpassing previous systems by 2%. Future work will focus on using a more diverse set of datasets for pre-training and exploring alternative selection mechanisms incorporating musical meter knowledge.

References

- Anton, J., Coppock, H., Shukla, P., and Schuller, B. W. Audio Barlow Twins: Self-Supervised Audio Representation Learning. sep 2022. URL <http://arxiv.org/abs/2209.14345>.
- Assran, M., Caron, M., Misra, I., Bojanowski, P., Bardes, F., Vincent, P., Joulin, A., Rabat, M., and Ballas, N. Masked Siamese Networks for Label-Efficient Learning. In *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, volume 13691 LNCS, pp. 456–473, apr 2022. ISBN 9783031198205. doi: 10.1007/978-3-031-19821-2_26. URL <https://arxiv.org/abs/2204.07141v1>.
- Assran, M., Duval, Q., Misra, I., Bojanowski, P., Vincent, P., Rabat, M., LeCun, Y., and Ballas, N. Self-Supervised Learning from Images with a Joint-Embedding Predictive Architecture. pp. 15619–15629, jan 2023. doi: 10.1109/cvpr52729.2023.01499. URL <https://arxiv.org/abs/2301.08243v3> <https://arxiv.org/abs/2301.08243>.
- Baevski, A., Zhou, H., Mohamed, A., and Auli, M. wav2vec 2.0: a framework for self-supervised learning of speech representations. In *Proceedings of the 34th International Conference on Neural Information Processing Systems*, 2020.
- Bao, H., Dong, L., Piao, S., and Wei, F. BEIT: BERT Pre-Training of Image Transformers. In *ICLR 2022 - 10th International Conference on Learning Representations*. International Conference on Learning Representations, ICLR, jun 2022. URL <https://arxiv.org/abs/2106.08254v2>.
- Bardes, A., Ponce, J., and LeCun, Y. VICReg: Variance-Invariance-Covariance Regularization for Self-Supervised Learning. In *The Tenth International Conference on Learning Representations, ICLR 2022, Virtual Event, April 25-29, 2022*. OpenReview.net, may 2022. doi: 10.48550/arxiv.2105.04906. URL <https://openreview.net/forum?id=xm6YD62D1Ub> <https://arxiv.org/abs/2105.04906>.
- Bertin-Mahieux, T., Ellis, D. P., Whitman, B., and Lamere, P. The million song dataset. In *Proceedings of the 12th International Conference on Music Information Retrieval*, 2011.
- Böck, S. and Davies, M. E. P. Deconstruct, analyse, reconstruct: How to improve tempo, beat, and downbeat estimation. In *International Society for Music Information Retrieval Conference*, 2020.
- Böck, S. and Widmer, G. Maximum filter vibrato suppression for onset detection. In *Proceedings of the 16th International Conference on Digital Audio Effects*, 2013.
- Böck, S., Davies, M. E. P., and Knees, P. Multi-task learning of tempo and beat: Learning one to improve the other. In *International Society for Music Information Retrieval Conference*, 2019.
- Bondarenko, Y., Nagel, M., and Blankevoort, T. Quantizable transformers: Removing outliers by helping attention heads do nothing. In *Advances in Neural Information Processing Systems*, volume 36, pp. 75067–75096, 2023.
- Böck, S. and Schedl, M. Enhanced beat tracking with context-aware neural networks. In *Proceedings of the 14th International Conference on Digital Audio Effects, DAFx 2011*, 09 2011.
- Caron, M., Touvron, H., Misra, I., Jegou, H., Mairal, J., Bojanowski, P., and Joulin, A. Emerging Properties in Self-Supervised Vision Transformers. In *Proceedings of the IEEE International Conference on Computer Vision*, pp. 9630–9640. Institute of Electrical and Electronics Engineers Inc., apr 2021. ISBN 9781665428125. doi: 10.1109/ICCV48922.2021.00951. URL <https://arxiv.org/abs/2104.14294v2>.
- Chen, T., Kornblith, S., Norouzi, M., and Hinton, G. A simple framework for contrastive learning of visual representations. In *Proceedings of the 37th International Conference on Machine Learning*, volume 119, pp. 1597–1607, 2020.
- Chen, X. and He, K. Exploring simple Siamese representation learning. In *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pp. 15745–15753. IEEE Computer Society, nov 2021. ISBN 9781665445092. doi: 10.1109/CVPR46437.2021.01549. URL <https://arxiv.org/abs/2011.10566v1> https://openaccess.thecvf.com/content_CVPR2021/papers/Chen_Exploring_Simple_Siamese_Representation_Learning_CVPR_2021_paper.pdf.
- Dangovski, R., Jing, L., Loh, C., Han, S., Srivastava, A., Cheung, B., Agrawal, P., and Soljacic, M. Equivariant self-supervised learning: Encouraging equivariance in representations. In *International Conference on Learning Representations*, 2022.
- Davies, M., Degara Quintela, N., and Plumley, M. Evaluation methods for musical audio beat tracking algorithms. Technical report, 2009.

- Davies, M. E. P. and Böck, S. Evaluating the evaluation measures for beat tracking. In *International Society for Music Information Retrieval Conference*, 2014.
- Desblancs, D., Lostanlen, V., and Hennequin, R. Zerone note samba: Self-supervised beat tracking. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 2023.
- Devlin, J., Chang, M.-W., Lee, K., and Toutanova, K. Bert: Pre-training of deep bidirectional transformers for language understanding. In *North American Chapter of the Association for Computational Linguistics*, 2019.
- Foscarin, F., Schlüter, J., and Widmer, G. Beat this! accurate beat tracking without dbn postprocessing. In *Proceedings of the 25th International Society for Music Information Retrieval Conference*, 2024.
- Gagnere, A., Peeters, G., and Essid, S. A Contrastive Self-Supervised Learning scheme for beat tracking amenable to few-shot learning. In *25th International Society for Music Information Retrieval Conference*, 2024.
- Gagnéré, A., Essid, S., and Peeters, G. Adapting pitch-based self supervised learning models for tempo estimation. In *ICASSP 2024 - 2024 IEEE International Conference on Acoustics, Speech and Signal Processing*, 2024.
- Gfeller, B., Frank, C., Roblek, D., Sharifi, M., Tagliasacchi, M., and Velimirovic, M. Spice: Self-supervised pitch estimation. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 28:1118–1128, 2020.
- Goto, M. AIST annotation for the RWC music database. In *International Society for Music Information Retrieval Conference*, 2006.
- Goto, M., Hashiguchi, H., Nishimura, T., and ichi Oka, R. Rwc music database: Popular, classical and jazz music databases. In *International Society for Music Information Retrieval Conference*, 2002.
- Gouyon, F. *A computational approach to rhythm description — Audio features for the computation of rhythm periodicity functions and their use in tempo induction and music content processing*. PhD thesis, Universitat Pompeu Fabra, 2006.
- Grill, J.-B., Strub, F., Altché, F., Tallec, C., Richemond, P., Buchatskaya, E., Doersch, C., Avila Pires, B., Guo, Z., Gheshlaghi Azar, M., Piot, B., kavukcuoglu, k., Munos, R., and Valko, M. Bootstrap your own latent - a new approach to self-supervised learning. In *Advances in Neural Information Processing Systems*, volume 33, pp. 21271–21284, 2020.
- Guzmán-rivera, A., Batra, D., and Kohli, P. Multiple choice learning: Learning to produce multiple structured outputs. In *Advances in Neural Information Processing Systems*, volume 25, 2012.
- Hadsell, R., Chopra, S., and LeCun, Y. Dimensionality reduction by learning an invariant mapping. *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2:1735–1742, 2006. ISSN 10636919. doi: 10.1109/CVPR.2006.100.
- Hainsworth, S. W. and Macleod, M. D. Particle filtering applied to musical tempo tracking. *EURASIP J. Adv. Signal Process.*, 2004(15):2385–2395, 2004.
- He, K., Chen, X., Xie, S., Li, Y., Dollár, P., and Girshick, R. Masked autoencoders are scalable vision learners. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 16000–16009, 2022.
- Henkel, F., Kim, J., McCallum, M. C., Sandberg, S. E., and Davies, M. E. P. Tempo estimation as fully self-supervised binary classification. In *ICASSP 2024 - 2024 IEEE International Conference on Acoustics, Speech and Signal Processing*, pp. 1356–1360, 2024.
- Hockman, J., Davies, M. E. P., and Fujinaga, I. One in the jungle: Downbeat detection in hardcore, jungle, and drum and bass. In *Proceedings of the 13th International Society for Music Information Retrieval Conference*, 2012.
- Holzapfel, A., Davies, M. E. P., Zapata, J. R., Oliveira, J. L., and Gouyon, F. Selective sampling for beat tracking evaluation. *IEEE Transactions on Audio, Speech, and Language Processing*, 20(9):2539–2548, 2012.
- Hung, Y.-N., Wang, J.-C., Song, X., Lu, W.-T., and Won, M. Modeling beats and downbeats with a time-frequency transformer. In *ICASSP 2022 - 2022 IEEE International Conference on Acoustics, Speech and Signal Processing*, 2022.
- Kong, Y., Lostanlen, V., Meseguer-Brocal, G., Wong, S., Lagrange, M., and Hennequin, R. STONE: Self-supervised Tonality Estimator. In *International Society for Music Information Retrieval Conference*, 2024.
- Kong, Y., Meseguer-Brocal, G., Lostanlen, V., Lagrange, M., and Hennequin, R. S-key: Self-supervised learning of major and minor keys from audio, 2025. URL <https://arxiv.org/abs/2501.12907>.
- Krebs, F., Böck, S., and Widmer, G. Rhythmic pattern modeling for beat and downbeat tracking in musical audio. In *International Society for Music Information Retrieval Conference*, 2013a.

- Krebs, F., Böck, S., and Widmer, G. Rhythmic pattern modeling for beat and downbeat tracking in musical audio. In *International Society for Music Information Retrieval Conference*, 2013b.
- Krebs, F., Böck, S., and Widmer, G. An Efficient State-Space Model for Joint Tempo and Meter Tracking. In *Proceedings of the 16th International Society for Music Information Retrieval Conference*, 2018.
- Lee, S., Purushwalkam Shiva Prakash, S., Cogswell, M., Ranjan, V., Crandall, D., and Batra, D. Stochastic multiple choice learning for training diverse deep ensembles. In Lee, D., Sugiyama, M., Luxburg, U., Guyon, I., and Garnett, R. (eds.), *Advances in Neural Information Processing Systems*, volume 29, 2016.
- Letzelter, V., Fontaine, M., Chen, M., Pérez, P., Essid, S., and Richard, G. Resilient multiple choice learning: A learned scoring scheme with application to audio scene analysis. In *Advances in Neural Information Processing Systems*, volume 36, pp. 6001–6013, 2023.
- Loshchilov, I. and Hutter, F. Decoupled weight decay regularization. In *International Conference on Learning Representations*, 2019.
- Lu, W., Wang, J.-C., Won, M., Choi, K., and Song, X. Spectnt: a time-frequency transformer for music audio. In *International Society for Music Information Retrieval Conference*, 2021.
- Lu, W.-T., Wang, J.-C., Kong, Q., and Hung, Y.-N. Music source separation with band-split rope transformer. *ICASSP 2024 - 2024 IEEE International Conference on Acoustics, Speech and Signal Processing*, pp. 481–485, 2023.
- Makansi, O., Ilg, E., Cicek, O., and Brox, T. Overcoming limitations of mixture density networks: A sampling and fitting framework for multimodal future prediction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019.
- Marchand, U. and Peeters, G. Swing Ratio Estimation. In *Proceedings of the International Conference on Digital Audio Effects*, 2015.
- MatthewDavies, E. P. and Böck, S. Temporal convolutional networks for musical audio beat tracking. In *27th European Signal Processing Conference*, 2019.
- McCallum, M., Korzeniowski, F., Oramas, S., Gouyon, F., and Ehmann, A. Supervised and unsupervised learning of audio representations for music understanding. In *Proceedings of the 23rd International Society for Music Information Retrieval Conference*, 2022.
- Nieto, O., McCallum, M., Davies, M., Robertson, A., Stark, A., and Egozy, E. The Harmonix Set: Beats, Downbeats, and Functional Segment Annotations of Western Popular Music. In *Proceedings of the 20th International Society for Music Information Retrieval Conference*, pp. 565–572, 2019.
- Niizumi, D., Takeuchi, D., Ohishi, Y., Harada, N., and Kashino, K. BYOL for Audio: Exploring Pre-Trained General-Purpose Audio Representations. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, pp. 1–15, apr 2022. ISSN 2329-9290. doi: 10.1109/taslp.2022.3221007. URL <https://arxiv.org/abs/2204.07402v2><https://dl.acm.org/doi/pdf/10.1109/TASLP.2022.3221007>.
- Quinton, E. Equivariant self-supervision for musical tempo estimation. In *Proceedings of the 23rd International Society for Music Information Retrieval Conference*, 2022.
- Raffel, C., McFee, B., Humphrey, E. J., Salamon, J., Nieto, O., Liang, D., and Ellis, D. P. W. mir_eval: A Transparent Implementation of Common MIR Metrics. In *Proceedings of the 15th International Conference on Music Information Retrievals*, 2014.
- Riou, A., Lattner, S., Hadjeres, G., and Peeters, G. Pesto: Pitch estimation with self-supervised transposition-equivariant objective. In *Proceedings of the 24rd International Society for Music Information Retrieval Conference*, 2023.
- Rupprecht, C., Laina, I., DiPietro, R. S., and Baust, M. Learning in an uncertain world: Representing ambiguity through multiple hypotheses. *2017 IEEE International Conference on Computer Vision (ICCV)*, pp. 3611–3620, 2016.
- Saeed, A., Grangier, D., and Zeghidour, N. Contrastive learning of general-purpose audio representations. pp. 3875–3879, 2020.
- Spijkervet, J. and Burgoyne, J. A. Contrastive learning of musical representations. In *Proceedings of the 22nd International Society for Music Information Retrieval Conference*, 2021.
- Su, J., Lu, Y., Pan, S., Wen, B., and Liu, Y. Roformer: Enhanced transformer with rotary position embedding. *Neurocomputing*, abs/2104.09864, 2024.
- Tzanetakis, G. and Cook, P. Musical genre classification of audio signals. *IEEE Transactions on Speech and Audio Processing*, 10(5):293–302, 2002.
- van den Oord, A., Li, Y., and Vinyals, O. Representation learning with contrastive predictive coding, 2019.

Wang, T. and Isola, P. Understanding contrastive representation learning through alignment and uniformity on the hypersphere. In *37th International Conference on Machine Learning, ICML 2020*, volume PartF16814, pp. 9871–9881. International Machine Learning Society (IMLS), may 2020. ISBN 9781713821120. doi: 10.48550/arxiv.2005.10242. URL <http://arxiv.org/abs/2005.10242>.

Zbontar, J., Jing, L., Misra, I., LeCun, Y., and Deny, S. Barlow Twins: Self-Supervised Learning via Redundancy Reduction. *38th International Conference on Machine Learning, ICML 2021*, mar 2021. doi: 10.48550/arxiv.2103.03230. URL <http://arxiv.org/abs/2103.03230>.

Zhao, J., Xia, G., and Wang, Y. Beat transformer: Demixed beat and downbeat tracking with dilated self-attention. In *Proceedings of the 23rd International Society for Music Information Retrieval Conference*, 2022.

A. Preliminary study

We evaluated the ability to detect the underlying hypotheses with different features. From an audio $\mathbf{x} = \{x_l\}_{l=1}^L$ we extract a sequence of features vector $\mathbf{V}^\mathbf{x} = [\mathbf{v}_1^\mathbf{x}, \mathbf{v}_2^\mathbf{x}, \dots, \mathbf{v}_T^\mathbf{x}] \in \mathbb{R}^{d_v \times T}$. The scoring function for a triplet $\mathcal{T}_k = \{A_k, \mathbf{P}_k, \mathbf{N}_k\}$ is defined as follows:

$$h_k(\mathbf{x}) = - \sum_{t_p \in \mathbf{P}_k} \log \frac{\exp(\text{sim}(\mathbf{v}_{A_k}^\mathbf{x}, \mathbf{v}_{t_p}^\mathbf{x})/\tau)}{\sum_{t_n \in \mathbf{N}_k} \exp(\text{sim}(\mathbf{v}_{A_k}^\mathbf{x}, \mathbf{v}_{t_n}^\mathbf{x})/\tau)}, \quad (7)$$

where τ is a temperature parameter and $\text{sim}(\mathbf{a}, \mathbf{b})$ denotes cosine similarity: $\text{sim}(\mathbf{a}, \mathbf{b}) = \frac{\mathbf{a}^\top \mathbf{b}}{\|\mathbf{a}\| \|\mathbf{b}\|}$.

In a preliminary study, we explored the accuracy of our scoring mechanism. Given annotated data we a fixed set of 20s long chunks. With ground truth annotations, we know the correct underlying sequence of PLP peaks corresponding to beats. The exact accuracy metrics correspond when we detect correct hypothesis. The octave one when we detect a metric that is acceptable. As an example let's say the correct is $\omega = 2$ and $\phi = 1$. Then when accounting for octave error $\omega = 4$ and $\phi = 1$ or $\phi = 3$ are also deemed correct. Finally the Metric level one account that a detection is correct when ratio ω is the same as the correct ratios.

Table. 3 gathers the different accuracy for mel spectrum, Chroma features computed either from Variable-Q-Transform (VQT) or Short Time Fourier Transform (STFT) and Mel-frequency Cepstral Coefficients (MFCC). For each of these feature we report the top k accuracies, $k \in \{1, 2, 3\}$ on three annotated datasets: Ballroom, GTZAN and SMC.

Table 3. Comparison of selection mechanism accuracy (in %) with different audio features.

TOP K ACCURACY	FEATURE	BALLROOM			GTZAN			SMC		
		EXACT	OCTAVE	METRIC LVL	EXACT	OCTAVE	METRIC LVL	EXACT	OCTAVE	METRIC LVL
1	MEL	19.5	82.7	20.7	20.9	85.2	22.8	20.2	62.8	31.8
	CHROMA STFT	14.3	85.3	14.8	17.3	88.1	18.4	14.0	66.7	22.5
	CHROMA VQT	22.0	82.3	22.5	23.8	85.5	24.8	24.8	65.9	31.0
	MFCC	6.2	71.5	10.1	8.4	88.2	12.0	5.3	60.5	11.2
2	MEL	37.5	92.9	40.6	34.9	95.1	38.2	31.0	80.6	44.2
	CHROMA STFT	46.3	93.1	48.5	<u>53.6</u>	<u>95.2</u>	<u>55.6</u>	45.0	79.8	53.5
	CHROMA VQT	<u>53.0</u>	<u>94.8</u>	<u>54.7</u>	<u>53.3</u>	<u>95.0</u>	<u>55.6</u>	<u>46.5</u>	<u>80.6</u>	<u>54.3</u>
	MFCC	21.1	84.0	34.1	32.6	92.3	44.9	22.1	78.4	41.9
3	MEL	50.2	96.4	56.2	55.2	97.3	62.2	44.2	86.0	59.7
	CHROMA STFT	63.8	95.8	68.1	73.0	97.2	77.1	64.3	85.3	78.3
	CHROMA VQT	69.6	96.7	72.7	69.3	97.3	73.0	59.7	86.8	66.7
	MEL	35.5	90.1	55.1	48.2	94.9	68.6	37.5	88.2	59.1

B. Cross Validation Metrics

We report in Table 4 and Table 5 the full cross-validation scores for beat and downbeat, respectively.

Table 4. Beat cross-validation scores. The best performances are shown in bold, and we have underlined the best-performing method for our proposed pre-training scheme.

PRE-TRAINING	FINE-TUNING	F1	CMLT	AMLT	F1	CMLT	AMLT	F1	CMLT	AMLT
		BALLROOM			BEATLES			HARMONIX		
WTA	BCE-DBN	97.2	95.7	<u>97.4</u>	94.0	87.0	91.4	95.4	89.2	94.3
WTA	ST-BCE	<u>97.5</u>	<u>96.1</u>	97.1	<u>94.6</u>	87.4	90.8	95.9	89.7	94.1
3-WTA	BCE-DBN	97.2	95.8	97.3	<u>94.5</u>	<u>87.8</u>	<u>91.7</u>	95.6	<u>90.1</u>	<u>94.5</u>
3-WTA	ST-BCE	97.3	95.7	96.7	94.4	<u>86.8</u>	91.1	<u>96.0</u>	89.8	93.5
SELF-TRAINING	BCE-DBN	97.9	96.9	97.7	96.2	91.7	92.9	96.9	92.5	95.6
SELF-TRAINING	ST-BCE	98.2	97.4	97.8	96.4	92.1	94.1	97.3	93.1	95.6
(FOSCARIN ET AL., 2024)		97.5	96.4	97.0	94.5	87.2	93.0	95.8	89.9	94.0
(BÖCK & DAVIES, 2020)		96.2	94.7	96.1	83.7	74.2	86.2	93.3	84.1	93.8
(HUNG ET AL., 2022)		96.2	93.9	96.7	94.3	89.6	93.8	95.3	93.9	95.9
(ZHAO ET AL., 2022)		96.8	95.4	96.6	0.0	0.0	0.0	95.4	90.5	95.7
		HAINSWORTH			RWC			SMC		
WTA	BCE-DBN	91.4	85.7	92.4	93.9	87.9	<u>94.6</u>	60.1	49.8	<u>65.0</u>
WTA	ST-BCE	91.5	84.6	90.9	93.9	87.2	<u>92.8</u>	61.7	46.8	<u>55.5</u>
3-WTA	BCE-DBN	90.6	83.9	<u>92.5</u>	94.3	88.6	94.4	59.8	<u>50.2</u>	64.7
3-WTA	ST-BCE	<u>92.8</u>	<u>87.1</u>	91.0	<u>94.4</u>	<u>89.2</u>	94.3	<u>62.1</u>	48.0	58.1
SELF-TRAINING	BCE-DBN	94.3	90.8	94.6	93.7	86.8	93.3	62.9	54.6	68.1
SELF-TRAINING	ST-BCE	94.8	90.4	94.5	93.5	86.0	93.7	64.2	52.9	63.7
(FOSCARIN ET AL., 2024)		91.9	84.0	90.9	96.1	90.1	93.6	62.7	51.4	61.0
(BÖCK & DAVIES, 2020)		90.4	85.1	93.7	0.0	0.0	0.0	55.2	46.5	64.3
(HUNG ET AL., 2022)		87.7	86.2	91.5	95.0	92.5	95.8	60.5	51.4	66.3
(ZHAO ET AL., 2022)		90.2	84.2	91.8	0.0	0.0	0.0	59.6	45.6	63.5

Table 5. Downbeat cross-validation scores

PRE-TRAINING	FINE-TUNING	F1	CMLT	AMLT	F1	CMLT	AMLT	F1	CMLT	AMLT
		BALLROOM			BEATLES			HARMONIX		
WTA	BCE-DBN	94.5	93.9	97.2	85.9	76.3	85.1	90.2	84.7	90.8
WTA	ST-BCE	94.8	91.6	94.5	86.6	67.9	78.6	90.3	79.7	85.0
3-WTA	BCE-DBN	94.2	93.5	97.2	87.4	77.3	86.4	90.5	85.7	91.2
3-WTA	ST-BCE	94.9	91.5	93.9	86.7	67.7	80.3	90.4	79.7	84.5
SELF-TRAINING	BCE-DBN	97.0	96.6	97.9	89.6	81.4	87.1	93.9	90.3	93.1
SELF-TRAINING	ST-BCE	96.6	94.3	96.1	89.4	74.4	82.2	93.1	84.5	87.7
(FOSCARIN ET AL., 2024)		95.3	92.9	94.1	88.8	73.8	82.0	90.7	81.2	85.9
(BÖCK & DAVIES, 2020)		91.6	91.3	96.0	0.0	0.0	0.0	80.4	74.7	87.3
(HUNG ET AL., 2022)		93.7	92.7	96.8	87.0	81.2	86.5	90.8	87.2	92.8
(ZHAO ET AL., 2022)		94.1	94.4	96.9	0.0	0.0	0.0	89.8	86.3	91.9
		HAINSWORTH			RWC			SMC		
WTA	BCE-DBN	78.6	73.4	86.2	92.7	89.1	93.6	—	—	—
WTA	ST-BCE	79.2	63.0	74.3	91.7	85.9	88.7	—	—	—
3-WTA	BCE-DBN	78.9	73.1	87.3	93.2	89.8	94.1	—	—	—
3-WTA	ST-BCE	80.1	62.1	74.6	92.9	88.1	90.9	—	—	—
SELF-TRAINING	BCE-DBN	83.2	78.6	88.9	92.4	87.8	92.7	—	—	—
SELF-TRAINING	ST-BCE	83.8	69.2	78.8	92.1	85.4	90.4	—	—	—
(FOSCARIN ET AL., 2024)		80.0	63.6	75.1	93.7	87.1	89.2	—	—	—
(BÖCK & DAVIES, 2020)		72.2	69.6	87.2	0.0	0.0	0.0	—	—	—
(HUNG ET AL., 2022)		74.8	73.8	87.0	94.5	93.9	95.9	—	—	—
(ZHAO ET AL., 2022)		74.8	71.2	84.1	0.0	0.0	0.0	—	—	—